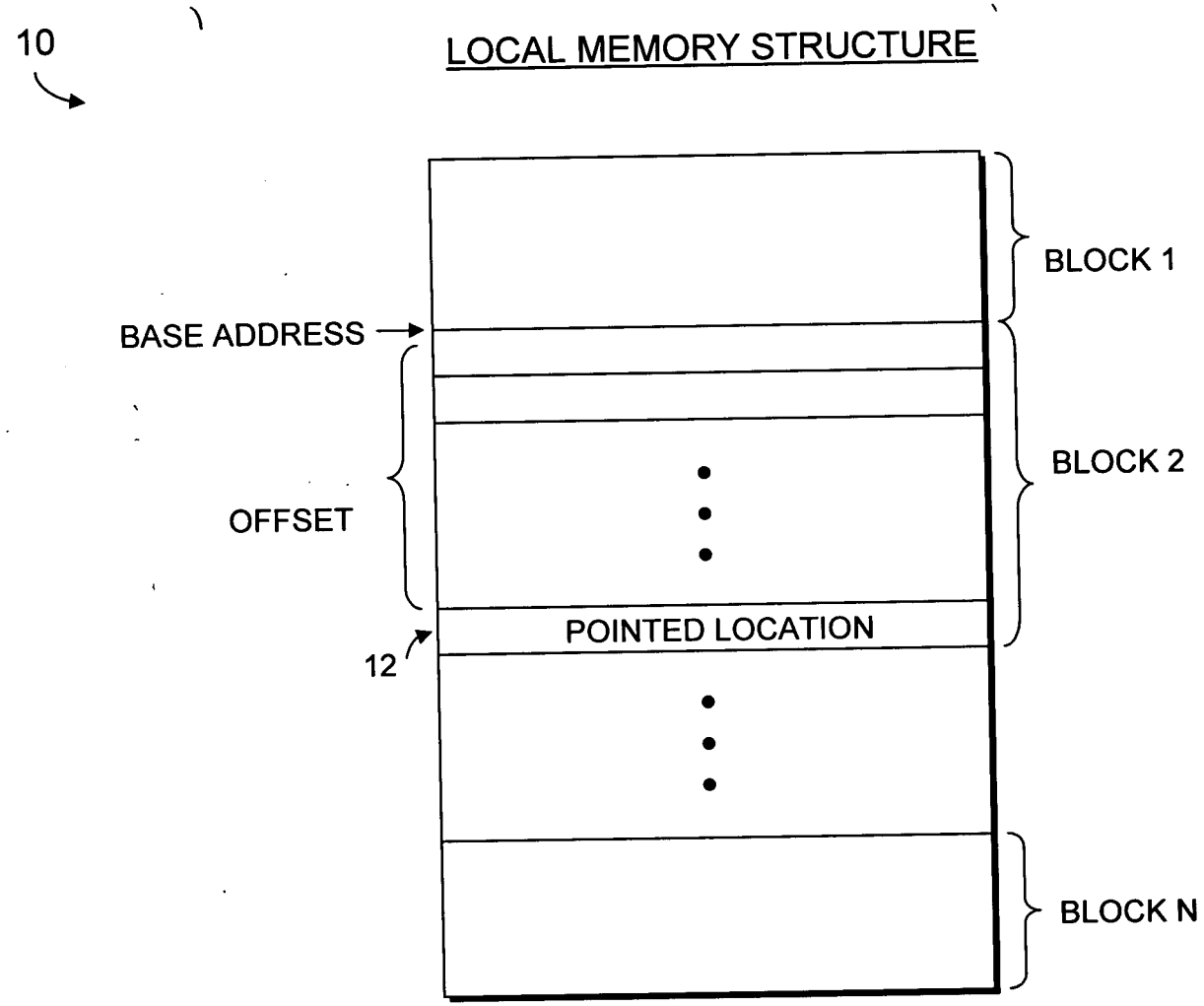
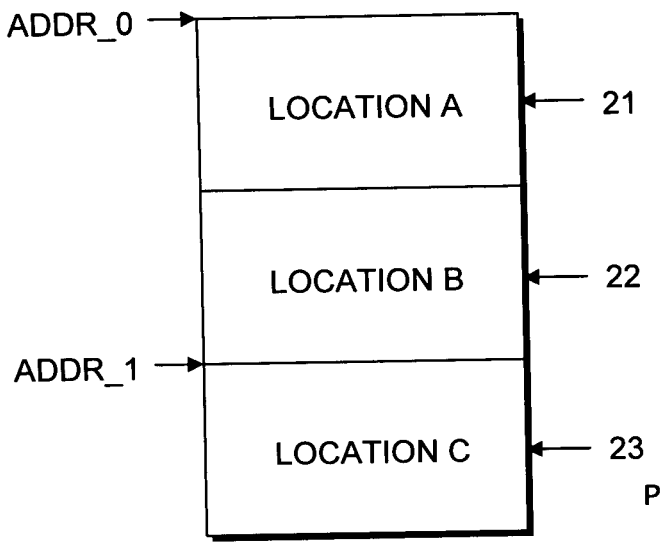


FIG. 1 (PRIOR ART)




```
1 Access A
2 Access B
3 Access A
4 Access C
5 Access A
6 Access C
```

Pseudo Code Sequence of accessing A, B, and C

```
base_addr_reg = Addr_0
Access A with offset 0
Access B with offset 1
Access A with offset 0
base_addr_reg = Addr_1
Access C with offset 0
base_addr_reg = Addr_0
Access A with offset 0
base_addr_reg = Addr_1
Access C with offset 0
```

Pseudo Code sequence of accessing A, B, and C after inserting code to Initialize the base address register

FIG. 2 (PRIOR ART)

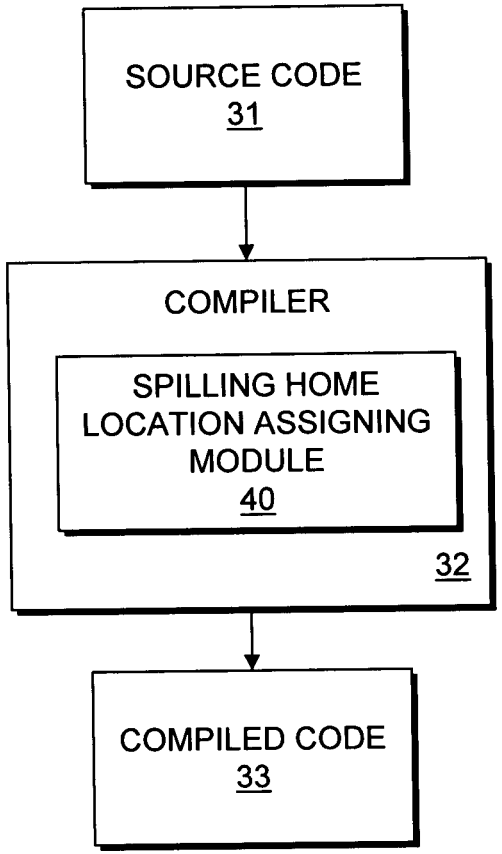

SOURCE CODE
31
COMPILER
SPILLING HOME LOCATION ASSIGNING MODULE
40
32
COMPILED CODE
33

FIG. 3

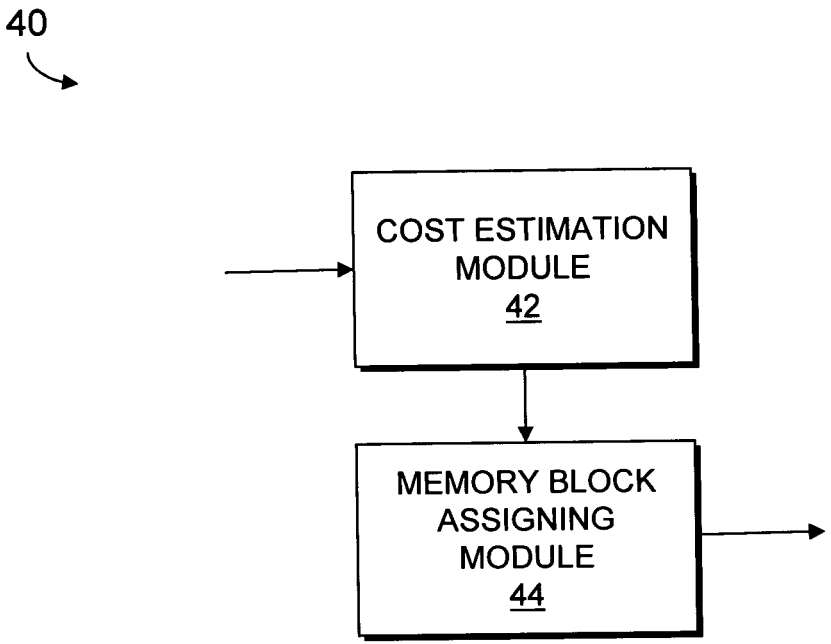
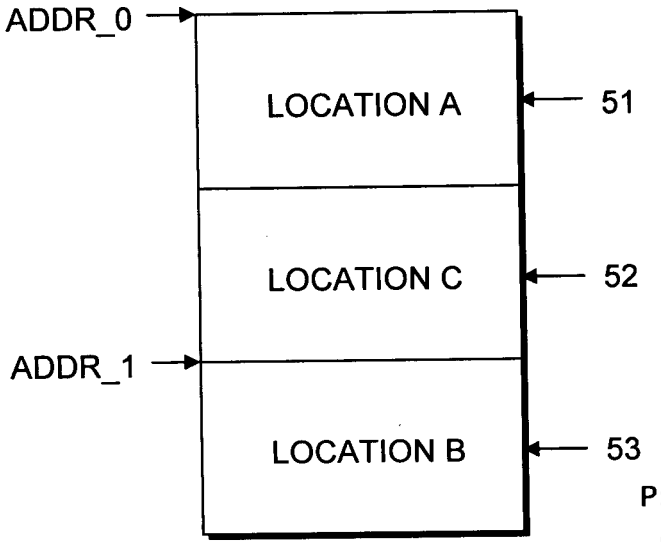

40
COST ESTIMATION MODULE
42
MEMORY BLOCK ASSIGNING MODULE
44

FIG. 4




```
1 Access A
2 Access B
3 Access A
4 Access C
5 Access A
6 Access C
```

Pseudo Code Sequence of accessing A, B, and C

```
base_addr_reg = Addr_0
Access A with offset 0
base_addr_reg = Addr_1
Access B with offset 0
base_addr_reg = Addr_0
Access A with offset 0
Access C with offset 1
Access A with offset 0
Access C with offset 1
```

Pseudo Code sequence of accessing A, B, and C after inserting code to Initialize the base address register

FIG. 5

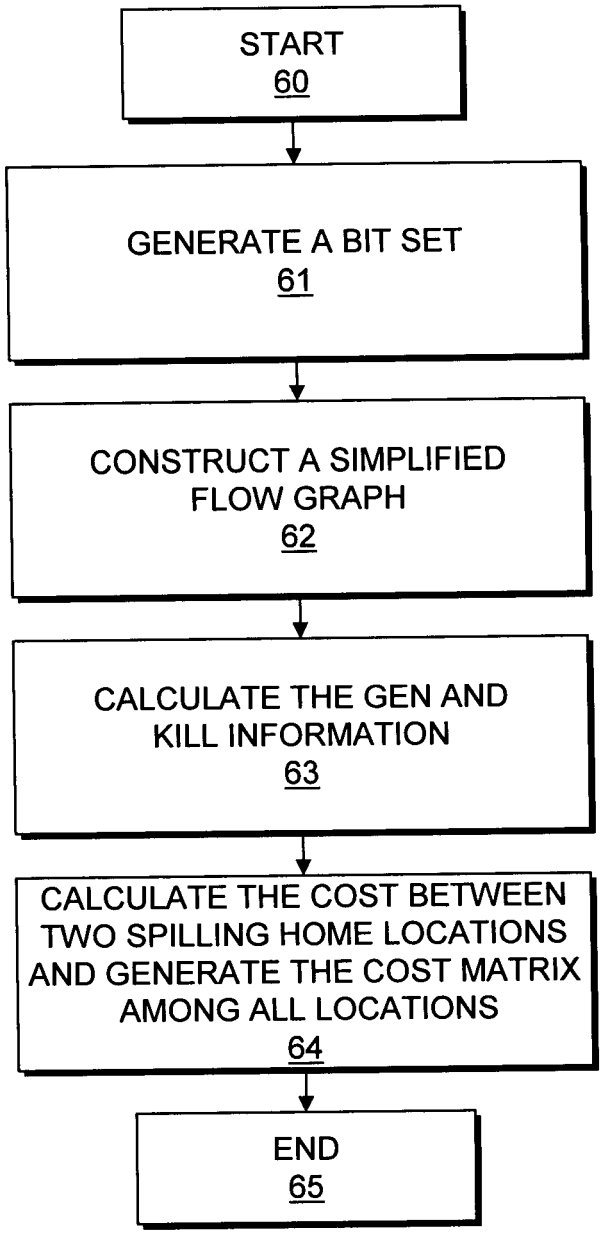

START
60
GENERATE A BIT SET
61
CONSTRUCT A SIMPLIFIED FLOW GRAPH
62
CALCULATE THE GEN AND KILL INFORMATION
63
CALCULATE THE COST BETWEEN TWO SPILLING HOME LOCATIONS AND GENERATE THE COST MATRIX AMONG ALL LOCATIONS
64
END
65

FIG. 6

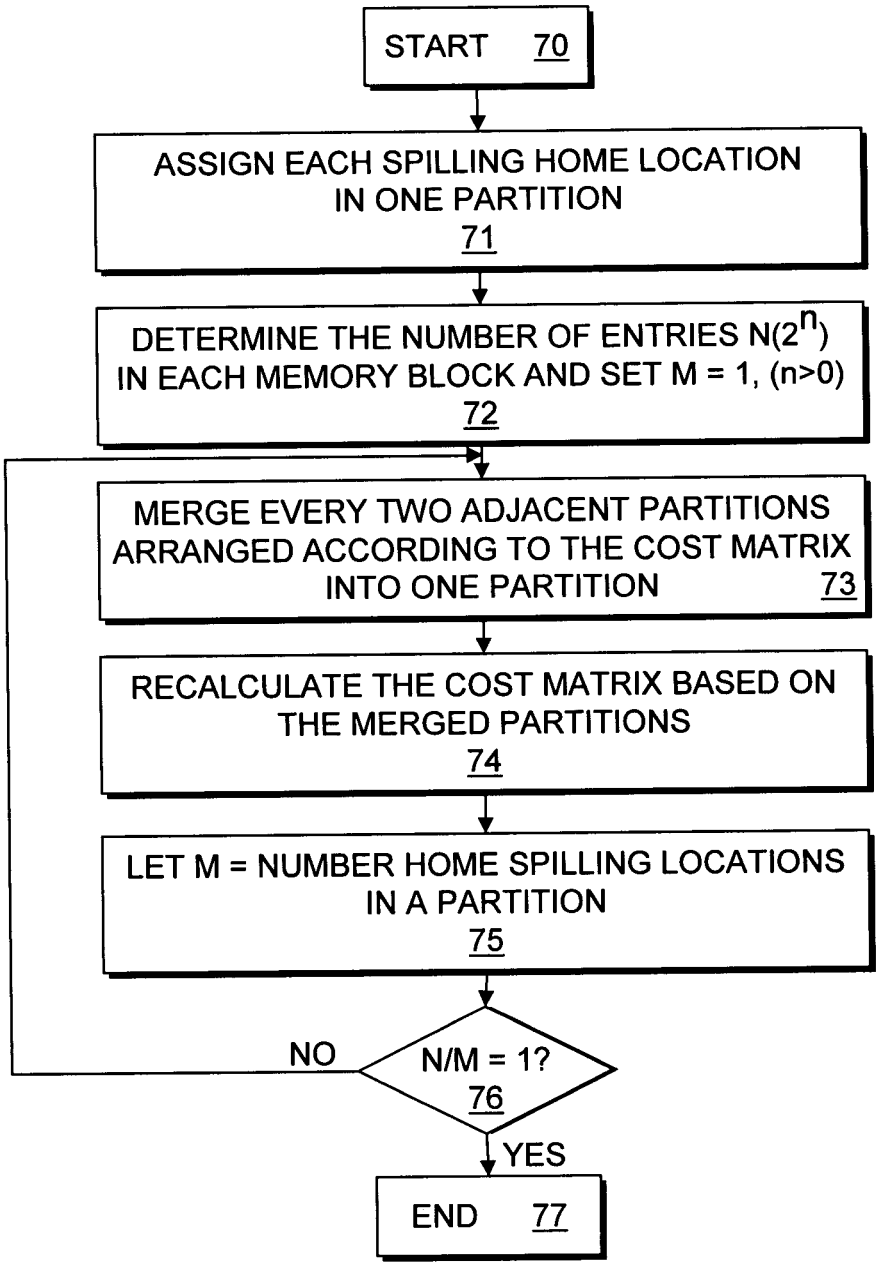

START 70
ASSIGN EACH SPILLING HOME LOCATION IN ONE PARTITION 71
DETERMINE THE NUMBER OF ENTRIES N(2^n) IN EACH MEMORY BLOCK AND SET M = 1, (n>0) 72
MERGE EVERY TWO ADJACENT PARTITIONS ARRANGED ACCORDING TO THE COST MATRIX INTO ONE PARTITION 73
RECALCULATE THE COST MATRIX BASED ON THE MERGED PARTITIONS 74
LET M = NUMBER HOME SPILLING LOCATIONS IN A PARTITION 75
N/M = 1? 76
NO
YES
END 77

FIG. 7

Iteration #	Number of partition after this iteration	Number of home locations in each partition after this iteration
1	M/2	2
2	M/4	4
3	M/8	8
...	...	...
n-1	$M/(2^{n-1})$	2^{n-1}
n	M/N	N

FIG. 8